Operational Expert System Applications in Canada

Edited by

Ching Y. Suen &
Rajjan Shinghal

PERGAMON PRESS

Operational Expert System Applications in Canada

Pergamon Titles of Related Interest

Titles in the Series

Cantu-Ortiz/ OPERATIONAL EXPERT SYSTEM APPLICATIONS IN MEXICO
Lee/ OPERATIONAL EXPERT SYSTEM APPLICATIONS IN THE FAR EAST
Liebowitz/ OPERATIONAL EXPERT SYSTEM APPLICATIONS IN THE UNITED STATES
Liebowitz/ PROCEEDINGS OF THE WORLD CONGRESS ON EXPERT SYSTEMS
Suen & Shinghal/ OPERATIONAL EXPERT SYSTEM APPLICATIONS IN CANADA
Zarri/ OPERATIONAL EXPERT SYSTEM APPLICATIONS IN EUROPE

Other Book Titles of Related Interest

Crespo/ REAL TIME PROGRAMMING
DeCarli/ LOW COST AUTOMATION COMPONENTS
Mladenov/ DISTRIBUTED INTELLIGENT SYSTEMS
Mowle/ EXPERIENCE WITH THE MANAGEMENT OF SOFTWARE PROJECTS
Reinich/ LARGE SCALE SYSTEMS
Rodd/ ARTIFICIAL INTELLIGENCE IN REAL TIME CONTROL

Journals

ANNUAL REVIEW IN AUTOMATIC PROGRAMMING
COMPUTER LANGUAGES
COMPUTERS & ELECTRICAL ENGINEERING
COMPUTERS & GRAPHICS
COMPUTERS & MATHEMATICS WITH APPLICATIONS
COMPUTERS & OPERATIONS RESEARCH
COMPUTING SYSTEMS IN ENGINEERING
ENGINEERING APPLICATIONS OF ARTIFICIAL INTELLIGENCE
EXPERT SYSTEMS WITH APPLICATIONS
MATHEMATICAL & COMPUTER MODELLING
MECHATRONICS
MICROELECTRONICS & RELIABILITY
NEURAL NETWORKS
PATTERN RECOGNITION

Operational Expert System Applications in Canada

Edited by
Ching Y. Suen & Rajjan Shinghal
Center for Pattern Recognition and Machine Intelligence
Concordia University

PERGAMON PRESS
Oxford • New York • Seoul • Tokyo

Pergamon Press Offices:

U.K. Pergamon Press plc, Headington Hill Hall,
Oxford OX3 OBW, England

U.S.A. Pergamon Press Inc. 395 Saw Mill River Road,
Elmsford, New York 10523, U.S.A.

KOREA Pergamon Press Korea, Room 613 Hanaro Building,
194-4 Insa-Dong, Chongno-ku, Seoul 110-290, Korea

JAPAN Pergamon Press, 8th Floor, Matsuoka Central Building,
1-7-1 Nishishinjuku, Shinjuku-ku, Tokyo 160, Japan

Copyright © 1991 Pergamon Press Inc.

All Rights Reserved. No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means: electronic, electrostatic, magnetic tape, mechanical, photocopying, recording or otherwise, without permission in writing from the publishers.

Library of Congress Cataloging-in-Publication Data

Operational expert system applications in Canada / (edited) by Ching Y. Suen and Rajjan Shinghal.
p. cm.
Includes index.
ISBN 0-08-040431-1
1. Expert systems (Computer science)--Canada. 2. Application software--Canada. I. Suen, Shing Y. II. Shinghal, Rajjan, 1945-

QA76. 76.E95063 1991
006.3 3 0971--dc20 91-11584
CIP

Printing: 1 2 3 4 5 6 7 8 9 10 Year: 1 2 3 4 5 6 7 8 9 0

Printed in the United States of America

The paper used in this publication meets the minimum requirements of American National Standard for Information Services -- Permanence of Paper for Printed Library Materials, ANSI Z39.48-1984

Contents

Preface		vii
1.	Diagnostic and Administrative Expert Systems at Bell Canada Network Services R. Douglas Bell	1
2.	A Knowledge-Based System for Configuration of Local Area Networks Lewis D. Baxter and David A. Faulkner	12
3.	The TRANSEPT Family of Expert Systems for the Preliminary Design of Power Networks F. D. Galiana, D. McGillis, I. Hafizullah, Q. D. Truong, and H. T. Pham	22
4.	Knowledge-Based and Object-Oriented Approaches to Process Planning at Northern Telecom Aldo Dagnino	32
5.	Health Expert Goes On-line Simon Freiwald, Ivan Zendel, David Benjamin, and Eliot Rubinov	45
6.	The Nervous Shock Advisor: A Legal Expert System in Case-Based Law Cal Deedman and J. C. Smith	56
7.	Expert System for Proposed Corporate Name Verification Maynard B. Hall, C. Anthony Harris, and Eugene Woo	72
8.	HIDES: The Highway Intersection Design Expert System Mark D. Brinsmead and James B. Tubman	80
9.	A Mix of Software Engineering and Knowledge Engineering Timothy Bult	100
10.	AMETHYST: A Multi-Expert Resource System for Public Sector Compensation and Benefits Personnel Kimiz L. Dalkir	114
11.	STATEX: An Expert Assistant for Statistical Analysis Joel Muzard, Eric Falardeau, and Michael G. Strobel	132
12.	Model-Based Automotive Diagnosis Using the Echidna Constraint Reasoning System William Havens, John Jones, Charlie Hunter, Stefan Joseph, and Afwarman Manaf	154
13.	Fuzzy Logic-Based Expert Systems for Operations Management I. B. Turksen	170
Author Index		184

This page intentionally left blank

Preface

Expert systems are now widely used in different parts of the world for various applications. In the past 4 years we have witnessed a steady growth in the development and deployment of expert systems in Canada. Research in this field has also gained considerable momentum during the past few years. A number of papers have been presented by Canadians at international conferences. Many seminar series and symposia on this subject have also been organized by both industrial and academic sectors. However, the field of expert systems is still quite young in Canada, and much more effort has yet to be devoted to it before we can make full use of this high technology. In the course of collecting papers for this book, we noticed great reluctance by several industrial groups in revealing and describing the types of expert systems they have developed or are using. It was only after an extensive effort that we gradually received a reasonable response from our colleagues in this community. But as is obvious, this book does not cover all the expert systems in Canada in 1990, but rather a representative subset of them. The expert systems described here were selected from those who responded to our invitation, which was extended to a large number of Canadian scientists and organizations active in expert systems and their applications.

This book contains 13 chapters contributed by 31 experts from universities and industries across Canada, covering a wide range of applications. These chapters are organized as follows.

The first two chapters present expert systems for applications in telecommunications. The first reviews expert systems that have been developed to provide consultations to field technicians for troubleshooting telephone equipment. The second presents a system for configuring local area networks to satisfy some given user requirements.

The next two chapters cover applications related to electric power and circuit boards. The third chapter discusses expert systems that have evolved for the preliminary design of high-voltage electric power networks, while the fourth chapter presents a system for assisting in the manufacture of printed circuit boards.

Related to health and medicine, Chapter 5 describes a system for advising users on how to reduce the risks of developing cancer.

The two chapters that follow are concerned with applications in the legal field. Chapter 6 discusses a system to advise a lawyer whether his client has grounds to sue someone for the tort of negligently inflicted shock. Chapter 7 presents an expert system to check the availability of a proposed corporate name based on the law about the proposed name being similar to an existing name, too general, or misrepresentative of the corporation's activities.

The next two chapters are related to land transportation and the electromechanical parts of a ship that operates near the North Pole. Chapter 8 discusses an expert system related to the design of intersections where three or four roads meet at the same level, with no underpasses or overpasses. Chapter 9 describes a system for detecting potential problems by analyzing the vibrations of pumps, compressors, fans, turbochargers, engines, winches, and motors on icebreakers.

Chapters 10 and 11 present expert systems that assist users in making decisions and interpreting data. The former describes a multiexpert system to assist human resource departments in carrying out hiring tasks. The latter discusses a system for interpreting the results of statistical analyses.

Chapter 12 describes an expert system for the diagnosis of automobile engine troubles. The last chapter discusses fuzzy logic-based systems for the management of operations such as production planning, scheduling, and inventory control.

The preparation of this book was supported in part by Bell Canada through a research contract related to the verification and validation of expert systems, and by research grants from the Natural Sciences and Engineering Research Council of Canada, the Ministry of Education of Quebec, and the Centre for Pattern Recognition and Machine Intelligence of Concordia University. We also wish to acknowledge the encouragement given by Professor Jay Liebowitz of George Washington University, the series editor of this book; Mr. Francois Coallier, Associate Director of the Division of Quality Engineering and Research of Bell Canada; the collaborations of our colleague Professor Peter Grogono and research associates Alan Bloch and Alun Preece; and the help offered by our secretary Irene Mazis. Finally, this project could not have been realized without the strong support of all the contributing authors who have devoted tremendous efforts to research and development of the field of expert systems in Canada and turning them into practical applications.

Ching Y. Suen
Rajjan Shinghal
Montreal, Canada
January 1991

Diagnostic and Administrative Expert Systems at Bell Canada Network Services

R. DOUGLAS BELL

Network Maintenance, London, Ontario, Canada

OVER THE PAST 3 years, a growing family of specialized diagnostic and administrative expert systems has been developed for the use of Bell Canada staff in Ontario, using commercially available expert system shells. While the early prototype systems were confined to PCs, the applications have spread in relevance, usage and platform; currently the same knowledge base developed on a PC is being accessed via a Province-wide network of some 3000 terminals, from a VAX mainframe. The total package currently amounts to some 6 megabytes of source code, and the largest application module to date embodies over 1,300 rules. Field and managerial acceptance has been enthusiastic; experience has indicated some unconventional coding techniques and extended usefulness. At present, diagnostic expert system modules offer field support to trained technicians in the repair of nine different types of hardware device; at least as many more such systems are in various stages of development. Less classical expert system applications include indices to the voluminous internal corporate documentation concerning policies, procedures, personnel benefits forms, technical notes and safety procedures. The usefulness and cost-effectiveness of this innovative technology has been amply demonstrated to the Bell community; usage of the package is spreading to neighboring provinces.

GENERAL BACKGROUND

The term *expert system* is applied to a computer program that incorporates and makes available to others the knowledge and experience of an expert in some specified field. Expert systems have in a few short years created a robust niche for themselves in industrial and commercial spheres, where the payoff for the work involved in developing them is frequently easy to demonstrate. For example, the difference between a newly trained technician and an expert repairman is the latter's accumulation of practical knowledge and experience, which can be transferred to the trainee either directly and face-to-face, or by encoding it in a widely available interactive program. As the experiences described in this paper demonstrate, this second approach is often more efficient and cost effective.

Major corporations noted for in-house implementation of expert systems include IBM (i.e., the Training Advisor program, which suggests professional staff development directions), DEC (the famous R1 computer system design consultant), Ford and General Motors (on-the-floor assembly-line robot maintenance, among others), E. I. du Pont (where ad hoc local expert systems have sprung up in many divisions and departments), and the Bell family of communications companies. Many of these endeavors have monopolized the effort of sizable professional programming teams for several years of development time before any rewarding product emerges; but, as demonstrated here, it is also feasible to implement useful systems in a matter of a few man-days, with the aid of appropriate shell packages. Some modern high-tech electronic hardware items come complete with their own intrinsic chip-

embedded diagnostic expert systems based on the same shells.

Diagnosis, or reasoning from a known set of symptoms to identify their (necessarily existing) common root cause, is one of the most popular (as well as most readily implemented) applications of expert system technology. Others include monitoring real-time on-line input data (such as from various sensors in an industrial process or hostile environment) and interpreting them by reference to predefined normal value sets, or the design, planning, control or optimized configuration of a presently nonexisting complex system (a much more demanding, and less prevalent, application, where in general it is difficult to guarantee the existence of a best solution).

BELL NETWORK SERVICES PROJECT

Before he had ever encountered the term *expert system,* the author's background in minicomputer maintenance crystallized in a BASIC-language program he developed in 1985 to interpret hexadecimal error messages for the benefit of nontechnical computer support staff, and advise them as to appropriate follow-up steps; this application was recognized a year later as a de facto expert system. In 1988 he was one of a small group of Bell staff who underwent training in the use of a commercial expert system shell package; they went on to develop and distribute a prototype diagnostic maintenance program for a particular piece of frequently repaired communications equipment, but this product never gained widespread acceptance because it ran only on PCs, which were inaccessible to most of the technical staff who could have used it.

That developmental effort was not entirely wasted, however, as the knowledge base was translated in 1989 into another shell environment which runs the same rule bases on PCs, Macintoshes, and VAX mainframes and can be queried from any VT100- or VT220-compatible terminals (of which some 3,000 are in use throughout the Bell Canada Network Maintenance system). That same year, field trials (including training of technicians and managers) proved the benefit of the approach, and stimulated nomination of 60 other device types for similar diagnostic treatment; subsequent managerial prioritization targetted 15 of these for early implementation.

It should be pointed out that the application area involves many different solid-state devices, which characteristically work properly for a long time between failures. Such failures are always catastrophic to a greater or lesser degree, since the purpose of the entire system is to maintain reliable, uninterrupted telephone service to Bell subscribers anywhere. A problem always requires urgent troubleshooting due to the pressure to restore full service as rapidly as possible. Although most technicians have been trained on most equipment, many months may elapse between their training on a particular device and the call to apply it; such a call could come at 2 a.m., demanding repairs to be made in 2 ft of snow, 2 hours' drive from civilization. The development of these expert systems is consequently need-driven!

The current project at Bell Canada has demonstrated that homegrown expert systems are capable of presenting troubleshooting consultation to a field technician as an expert would, and drawing logical conclusions (including spotting when the user is hopelessly lost, and referring him to a support hot line); the interactive systems support revision of user input responses, storage of the current status of an inquiry for later continuation (especially useful when attempting to alleviate intermittent malfunctions), on-line context-sensitive help to the user, and compatibility with most existing network maintenance terminals.

The proven benefits enjoyed as a result of implementing these diagnostic expert systems include:

1. upgrading the performance of technicians to expert level;
2. reinforcing previous technical staff training;
3. extending technical support availability to

24 hours per day, while drastically decreasing demands on support personnel and minimizing the possible embarrassment entailed in having to ask a colleague for assistance; and

4. decreasing downtime of crucial systems by expediting prompt repairs and maintenance.

Additional benefits uncovered in the present shell approach (as opposed to *de novo* programming) include a much faster learning curve, as well as ease of updating the knowledge base and incorporating preexisting text, such as corporate engineering and technical documentation.

DEVELOPMENT PROCESS

Problem Selection

The technical literature in the expert systems field is fairly consistent in recommending criteria for evaluating the appropriateness of investing developmental effort in expert systems. One looks for:

1. a well-defined, high-profile subject area (one where frequent problems motivate such treatment), in which only a low level of expertise is generally available;
2. the availability of
   a. a cooperative and committed subject matter expert,
   b. a competent project developer, and
   c. hardware and software tools adequate to render the project potentially practicable; and
3. a high level of management support.

In the present case, as mentioned above, once the pioneering module was up and running and demonstrated the potential usefulness of the application, there was little difficulty in identifying subsequent target areas; the challenge was more to decide which were most urgent.

One consistent theme characterizing the project to date is the recurring realization of the potential for applying the expert system approach to new topics, on the part of both field users and the developer: "If it can do X, why not Y, Z, and A as well?" Programmers themselves frequently come up with unorthodox new ideas from trying innovative angles indicated by perceived need, such as incorporating text-heavy technical document retrieval and personnel benefit forms handling into the package. After all, they are also domains of isolated expert knowledge of great potential usefulness to the Bell community at large, if only some method can be found to make them more widely accessible. A seemingly unrelated application, recently developed under considerable demand, is a schematic presentation of standardized corporate safety procedures (starting out with "Whom to Call in Case of an Accident"), which turns out to lend itself nicely to the rule-based expert system approach. (See Figure 1.1.)

Knowledge Acquisition Process

In the Bell Canada environment, it is generally not difficult to identify the unique expert in the maintenance of any particular piece of equipment; in at least one prominent case, that individual was facing retirement, which strengthened the motivation to capture his expertise for the ongoing benefit of the staff and user community at large. During the development of the first half-dozen diagnostic systems, a stable knowledge acquisition methodology has been refined; this process characteristically transpires during a series of direct interviews, generally limited to 2 hours at a stretch, each of which may give rise to an average of 3 days of implementation work followed by some 4 hours of testing the expanded prototype.

Initially the potential scope of the application is mapped out by applying the traditional "divide-and-conquer" analytical approach to the possible array of malfunction symptoms. This hierarchical analysis systematically rules out irrelevant hypotheses and focuses on the essence of the problem. At the

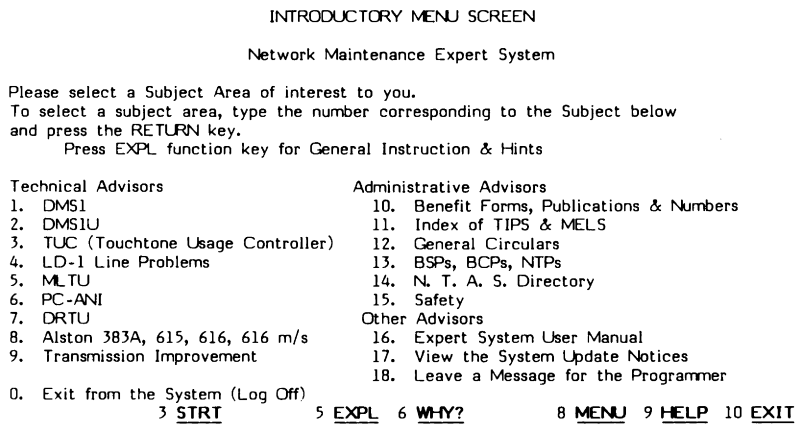
INTRODUCTORY MENU SCREEN
Network Maintenance Expert System
Please select a Subject Area of interest to you.
To select a subject area, type the number corresponding to the Subject below
and press the RETURN key.
Press EXPL function key for General Instruction & Hints
Technical Advisors
1. DMS1
2. DMS1U
3. TUC (Touchtone Usage Controller)
4. LD-1 Line Problems
5. MLTU
6. PC-ANI
7. DRTU
8. Alston 383A, 615, 616, 616 m/s
9. Transmission Improvement
0. Exit from the System (Log Off)
Administrative Advisors
10. Benefit Forms, Publications & Numbers
11. Index of TIPS & MELS
12. General Circulars
13. BSPs, BCPs, NTPs
14. N. T. A. S. Directory
15. Safety
Other Advisors
16. Expert System User Manual
17. View the System Update Notices
18. Leave a Message for the Programmer
3 STRT 5 EXPL 6 WHY? 8 MENU 9 HELP 10 EXIT

FIGURE 1.1. Introductory menu screen. Undocumented numeric options (such as 19 or 20) allow system users who are aware of their existence to access incomplete modules still under development. The masks along the bottom margin of the screen image are labels for function keys.

same time, some detailed example cases are analyzed from the expert's experience. On the basis of the preliminary decision tree a prototype program is devised of limited breadth and depth, but sufficient to illustrate the approach to the domain expert; flow charts are also frequently developed (and sometimes even provided by the expert) to utilize the common (but often overlooked) eidetic analytical skills of such experts. This method of presentation avoids imposing the incomplete series of computer screen displays on them and offers conceptual feedback for detailed evaluation. The ultimate goal of the knowledge acquisition process is to produce a decision tree that embodies most known patterns of malfunction, and identifies the most appropriate repair tactic for each. The final tree is approximated through several iterations of refinement.

What is being acquired here is heuristic knowledge in terms of high-level rules of thumb, which efficiently relate symptoms to their most likely causes rather than deep causal-level knowledge of electronic device behavior. This approach to modeling the diagnostic problem has proven not only quite reliable, but much more efficient in terms of rapid convergence on the defective subunit than the more instrumentation-intensive alternative of logical testing of electronic signals at all pertinent nodes of the circuitry. The principle throughout the development process is that the expert knows the *shortest* way to arrive at and fix the problem, by experience, which is predictably faster than thumbing through technical manuals on the spot.

In outline, then, the knowledge acquisition and development process is as follows:

1. initial interview and scoping of the application;
2. construction of a demonstration prototype based on a subset of the domain;
3. review of the prototype with experts (for relevance) and users (for acceptability);
4. further knowledge acquisition interviews and refinement and/or expansion of the prototype system; and
5. iteration of steps 3 and 4 until the prototype is deemed acceptable, after which it becomes an operational knowledge base.

This procedure is essentially analogous to the conventional rapid prototyping method, used here in expert system development instead of standard software engineering.

It should be pointed out in this context that highly qualified technical experts exhibit a wide range of degrees of insight into, and ability to express verbally, their knowledge and problem-solving techniques; the knowl-

edge engineer must employ whatever communication strategy proves most effective in a given personal relationship, constantly bearing in mind that his program development project is dedicated to serving the needs of the expert and his colleagues, and not vice versa! The comfort and interest of the expert must be maintained throughout the protracted early development steps, largely by both dealing appropriately with his subjective ego needs and providing him with rapid feedback to each interview session.

In designing the text windows selected for display to the questioning user, presenting the actual words of the expert to the client community has proven beneficial in enhancing comprehension of the concepts, content, and direction of thought. A cultural choice must be constantly made as to whose jargon is to be employed, with the obvious choice being that of the field engineer, not the programmer.

Tools commonly employed to support the knowledge acquisition process include a standing easel or padboard, for large-scale sketching of entities in the system under consideration and their relationships, and an electronically rotating padboard that dumps an image of the flowchart to FAX paper. The same ends can equally be met by using a plain pad of paper; in this case, the expert watches and verifies the knowledge engineer's recording of each concept, and pages are numbered to maintain conceptual relationships. Neither audio nor video recording has been used, largely to avoid intimidating senior employees.

During the development of a few diagnostic modules, multiple experts were involved; they characteristically revealed complementary sets of partial knowledge, none of which were necessarily invalid. For example, regional disparities surfaced with regard to optimal testing methodology; both were pertinent given certain scenarios, and the ultimate resolution stemmed from responses to questions such as "Under which circumstances might you use the *other* approach?" In one case it proved necessary to segregate the knowledge base into shortcut and detailed reasoning paths, presenting the faster approach first.

Knowledge Representation

The shell tool in use, Level 5 from Information Builders, has proven adequate for rapid prototyping of diagnostic expert systems (although it might not suffice for other, more demanding areas of expert system application, such as configuration, pattern recognition, and qualitative modeling). It represents expert knowledge in the common form of sets of logical production rules of the form IF A AND B AND NOT C THEN D, which serve as grist to the built-in inference engine. (Other commercial shells implement frame or semantic net representations, or some hybrid of these.) Level 5 supports Boolean logical variables, whose names can be up to 60 characters in length for readability, handles single or multiple-choice user input, performs indicated arithmetic computations, and allows access to external databases (although this feature has not yet been used in the current project). An earlier generation of the same product, marketed as Insight Plus, has resulted in the development of hundreds of applications at E. I. du Pont.

After achieving poor speed performance (an order of magnitude too slow) from an early knowledge base coded as short rules (IF A THEN B, IF B THEN C, IF C THEN D), the author established the policy of encoding an entire chain of inference in each rule; thus, most rules present a visible conclusion to the user, without a plethora of intermediate variables. The resulting rule form, IF A AND B AND C THEN GOAL AND DISPLAY CONCLUSION, avoids both reevaluating the variables A, B, and C, and also tracing an involved chain of inference before reaching the conclusion; this implies that any single symptom may appear in a number of rules, but the redundancy at the source-code level is rewarded with greatly improved performance. This approach to knowledge base de-

sign is perhaps at odds with the classical tenets of structured programming, and may illustrate a frank paradigm shift between procedural and declarative coding methodologies; it is conceivable that an inference engine of another design might perform more respectably when presented with numerous short rules and intermediate variables.

An essential principle of knowledge base design is that rules must clearly express the intended expert logic at a high level of representation. Whereas the initially chosen shell package insisted on rather cryptic parameter names, Level 5 allows much longer labels, such that each rule's values and parameters are expressed in the form of complete English sentences. The rule exhibited in Figure 1.2A demonstrates a transitional step between the terse and verbose coding approaches, while that in Figure 1.2B is entirely verbose. This style standard has been found to significantly enhance accuracy and maintainability of the rule base, as well as general readability.

Other benefits of using Level 5 include the possibility of linking separate knowledge bases, so that there is no effective limit on the size of a program, and the ease of preparing common modular subunits (e.g., a uniform user interface between different applications), expedited by the sharing of parameters between knowledge bases. So far, in the applications under discussion, numerical confidence values (and their attendant computational complexity) have not been utilized, because the knowledge expressed by the experts consists of well-defined Boolean information (as opposed to problems of fuzzy or missing data which are common in other domains, where the evolution of the subdisciplines of truth maintenance and nonmonotonic logic has been required). To demonstrate the ease of use of Level 5, only a dozen or so key words of the shell language are required to create an application of this sort. Although an explanatory trace of logic flow is supported in Level 5, it has proven to be essentially meaningless to the field users, few of which have any programming background; in the present context it would require massive rewording and commenting of the rule

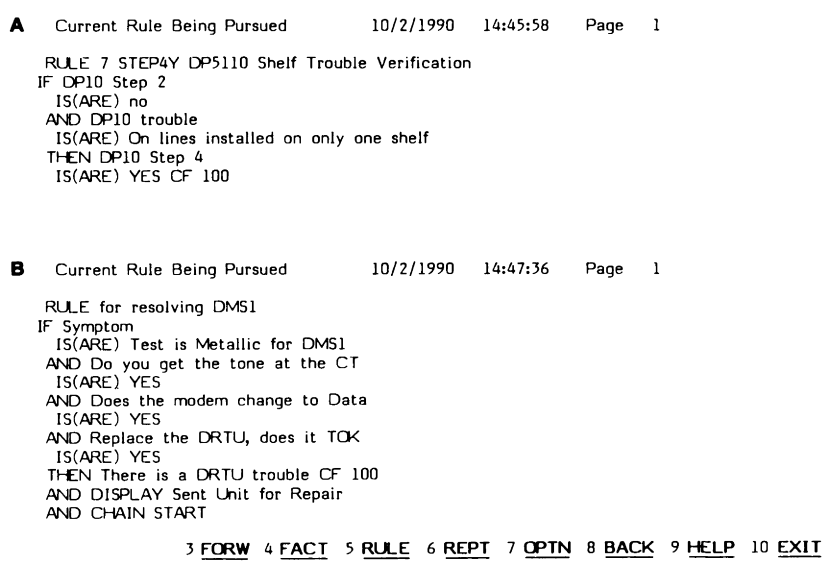
A Current Rule Being Pursued 10/2/1990 14:45:58 Page 1
RULE 7 STEP4Y DP5110 Shelf Trouble Verification
IF DP10 Step 2
IS(ARE) no
AND DP10 trouble
IS(ARE) On lines installed on only one shelf
THEN DP10 Step 4
IS(ARE) YES CF 100
B Current Rule Being Pursued 10/2/1990 14:47:36 Page 1
RULE for resolving DMS1
IF Symptom
IS(ARE) Test is Metallic for DMS1
AND Do you get the tone at the CT
IS(ARE) YES
AND Does the modem change to Data
IS(ARE) YES
AND Replace the DRTU, does it TOK
IS(ARE) YES
THEN There is a DRTU trouble CF 100
AND DISPLAY Sent Unit for Repair
AND CHAIN START
3 FORW 4 FACT 5 RULE 6 REPT 7 OPTN 8 BACK 9 HELP 10 EXIT

FIGURE 1.2. (A) Vestigial terse rule, difficult to maintain, explain, or update. (B) Rule using verbose parameter names, much clearer to understand. Both are displayed by Level 5's rule editor environment, with different function key labels (here represented only once).

base to render this output useful. The same function is better served in this case by reviewing the user's input stream for the current problem. The author found it necessary to develop a utility for maintaining an inventory of labels and checking for duplicate names; additional debugging tools could be extremely useful during the development of larger applications, and even running the ASCII rule base source file through an external spelling checker has proven helpful.

Knowledge Testing and Evaluation

In such a practical field, on whose accuracy depends the behavior of an entire communications system, the ultimate criteria of knowledge quality are how well and reliably the system works and how effectively the diagnostic advisor minimizes downtime. Beyond the early prototype phase, where the subject matter expert reviews numerous specific cases (selected by brute force, not random sampling) for appropriateness, the ongoing integrity of the knowledge base is served mainly by feedback from the field use community; in this situation, rapid response to their observations, comments and suggestions is essential to maintain the requisite trust in the product.

IMPLEMENTATION

Maintenance Strategies

At present, about 5% of the developer's time is being spent on maintenance of existing modules; this effort is largely concerned with adding new rules, as opposed to correcting existing rules. There is a growing need for better logical tools, for instance, to perform logical checks of the knowledge base. For example, of the next dozen projects slated for implementation, four or five are estimated to be quite large, complex systems requiring over a thousand rules each (and perhaps 4 to 6 months of development time). The existing modules run to some 6 megabytes of source code; so far only one application has exceeded a thousand rules. Whereas for smaller systems each subset of the decision tree may be able to be tested conceptually with the expert during the iterative cycle of expanding through adjacent subject areas, in larger applications such as these it will be much more difficult to test all eventualities. In this context an automatic validation tool that could identify missed cases (sets of permissible inputs for which no output is defined) would be of great benefit. Already, in the pioneering DMS Advisor, a matrix-represented truth table was implemented (originally in a parameter vs. value spread sheet, which was later transformed programmatically into a set of rules); there are 23 different alarm lamps in the DMS device, almost any combination of which might conceivably be observed in some fault state, and this type of input is too complex to be handled adequately by a flowchart designed for human perusal.

Figure 1.3(A–E) presents a brief walkthrough of a DMS diagnostic session, demonstrating single and multiple user inputs per step, and incorporating Level 5's arithmetic computation capability; the conclusion in Figure 1.3E is a mathematical function of the response(s) in 1.3D.

When problems have arisen in using the diagnostic systems in the field, it has proven helpful for the developer and the technical user (or manager and field trainee) to log onto the package while in telephone contact, so that they can walk through the consultation dialogue together, confident that they are looking at the same screen and choosing the same options to proceed through the analysis. In this way, a screen dump of the troublesome text sections in question allows a quick search through the ASCII rule base to identify misunderstandings.

Rules are ordered into groups in the source code by subject domain, and commented for easy identification. The shell supports ready updating of the knowledge base whenever the expert suggests enhancements (new problems or previously overlooked details) or field users

A

DMS 1 MAINTENANCE ADVISOR

What is the current condition of the system?

System was Down, and either restarted automatically or manually

SYSTEM DOWN

System appears to be OK, but you have trouble reports

ALARM

Shelf Turn Up Problem

SLTE, Testing Problems

Return to Expert System Main Menu

B

DMS 1 MAINTENANCE ADVISOR

At the CT examine the QPP421 Alarm Office Board. Verify that the MTCE/ALM switch is in the ALM position and that the rotary switch on the bottom of the board is in the CCT position. Press the RESTART and RESTART ARM buttons simultaneously on the QPP431 to restart the controller. This will assist in determining if the alarm is a solid fault.

Please select ALL of the lamps that are now lit, INCLUDING the RCT and CCT indicators. See Help for shortcuts in moving through the list of lamps.

1. DGP FAIL A
2. DGP FAIL B
3. LINE FAIL A
4. LINE FAIL B
5. LINE FAIL P
6. BYPASS OP

- - - - - - - - - - - - - - - - -

17. FAULT LOC
18. LL DET FAIL
19. SYS CONT
lamp 20
lamp 21
lamp 22
lamp 23
lamps 1 to 23 are NOT lit
RCT 1
RCT 2

After making your selections press F4 for DONE

1 PAGE 3 STRT 5 EXPL 6 WHY? 8 MENU 9 HELP 10 EXIT

FIGURE 1.3. Sample diagnostic dialogue of DMS 1 maintenance advisor (boldface and underscoring indicate options chosen, which actually appear in inverse video on the screen). (A) Introductory menu screen, expecting one selection. (B) Query screen expecting any number of selections as input (actually requires three screen images to complete the list, indicated by dashed lines). (C) Additional query, with instruction and feedback. (D) Repeated multiple selection screen(s), similar to (B). (E) end of the line for this troubleshooting diagnosis. A hexadecimal computation based on the pattern of alarm lights activated identifies which Line Card on which Shelf is the locus of the problem indicated.

locate inconsistencies or inaccuracies. Furthermore, it is not unheard of for the experts and users to identify errors in the technical documentation that accompanied the equipment under discussion. As a result of such requirements for updates, rules are easily added; in actual experience, modification of existing rules is rarely required.

Acceptance and Justification

So far, some 400 out of a potential 2,000 field technicians have undergone training to use the diagnostic package; the response, both before, during, and afterward was predictably varied. There was a small degree of cyberphobia, but remarkably little evidence of per-

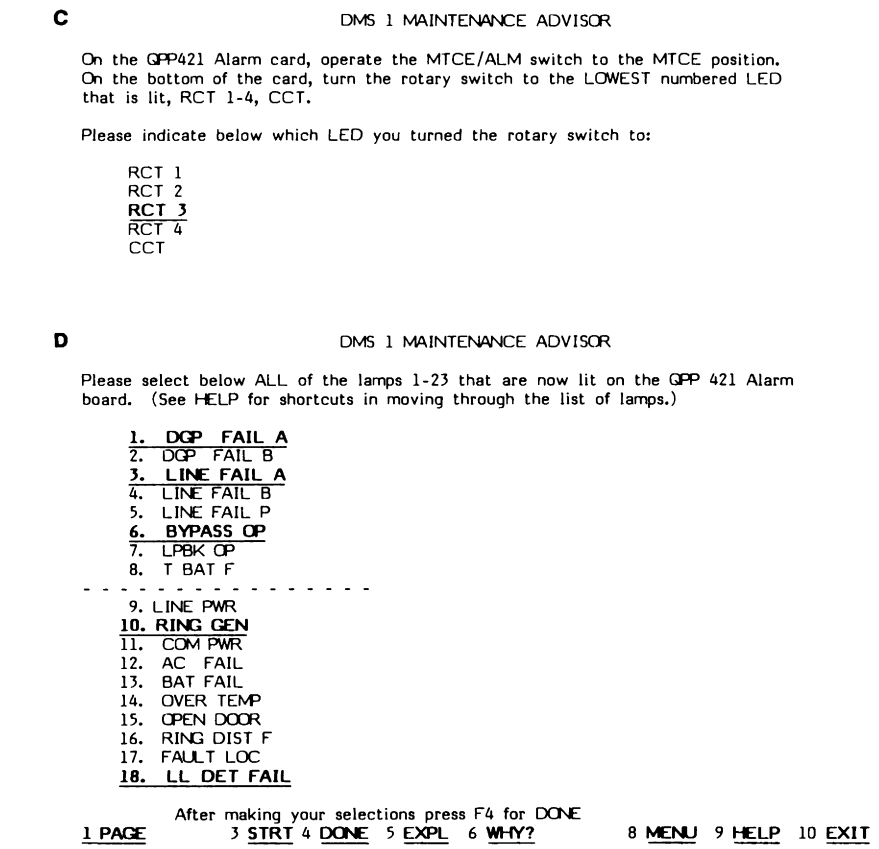
C
DMS 1 MAINTENANCE ADVISOR
On the GPP421 Alarm card, operate the MTCE/ALM switch to the MTCE position. On the bottom of the card, turn the rotary switch to the LOWEST numbered LED that is lit, RCT 1-4, CCT.
Please indicate below which LED you turned the rotary switch to:
RCT 1
RCT 2
RCT 3
RCT 4
CCT
D
DMS 1 MAINTENANCE ADVISOR
Please select below ALL of the lamps 1-23 that are now lit on the GPP 421 Alarm board. (See HELP for shortcuts in moving through the list of lamps.)
1. DGP FAIL A
2. DGP FAIL B
3. LINE FAIL A
4. LINE FAIL B
5. LINE FAIL P
6. BYPASS OP
7. LPBK OP
8. T BAT F
9. LINE PWR
10. RING GEN
11. COM PWR
12. AC FAIL
13. BAT FAIL
14. OVER TEMP
15. OPEN DOOR
16. RING DIST F
17. FAULT LOC
18. LL DET FAIL
After making your selections press F4 for DONE
1 PAGE 3 STRT 4 DONE 5 EXPL 6 WHY? 8 MENU 9 HELP 10 EXIT
E
DMS 1 MAINTENANCE ADVISOR
Replace Line Card: 5 , on Shelf: 4 at the CCT. After the card is replaced, press the RESTART and RESTART ARM buttons simultaneously on the GPP 431 to see if the change clears the fault.
Call Technical Support as required for assistance.
Press the RETURN key to return to the Expert System Main Menu.

FIGURE 1.3. Continued.

ceived threat from the program; although some expressed a preference for a user's manual (22 pages, mostly an introduction to using Level 5) in lieu of training sessions, others shooed the trainers away from the terminals in their eagerness to play with the system, and were quickly gratified with its potential profitability.

As is expected with troubleshooting systems of this sort, the level of usage of any individual is expected to drop off with time, as he or she acquires the knowledge embedded in the system and, thus, has less need for the program. In this context, cost–benefit analysis based solely on frequency or prevalence of use will result in an underestimate of the system's justification. The system is presently logging some 400 accesses monthly, some of which may be for educational "playing around" or self-testing; but many are clearly prompted by the rare and geographically scattered malfunctions that are the raison

d'être of both the network maintenance service and the expert system package.

Even though all possible troubles are not incorporated into the knowledge base, the package serves quite effectively to lead the technician to the general area of the problem, where he can apply his own intelligence and training most efficiently. Rather than replacing or obviating the company's technical training courses (which may have been last updated several months before a relevant malfunction occurred), the expert system amplifies and augments them. From the beginning, the tools in this package were uniformly presented to and received by the field technicians as "their system," and were focused on ease of use and relevance to the technicians' needs, instead of those of management. Feedback for improvement and modification has been eagerly solicited; in time an on-line message facility was created with this end in mind, although in fact it has generally been used more for suggesting other beneficial subject areas than for identifying inaccuracies or incompleteness in existing modules. The occasional detailed recommendations for changes in text or logic design have met with prompt implementation and personal response. While technicians sometimes express surprise at the logic or balance of some specific part of the knowledge base, they are generally educable and appreciative of the new tool. One key contribution to widespread acceptance has been the absolute avoidance of the term *Artificial Intelligence;* in the present environment it carries a most distasteful, science-fiction connotation and suggests robots replacing human operators. Another is a conscious attempt to minimize consultation duration by structuring the decision trees to use query screen nesting as shallow as practicable. In the present package of modules, depths of query screen nesting vary from 5 to 16.

Graphics

Other shell packages support graphical display of the rule set, a feature that has proven extremely beneficial in many applications. Although one might have desired a generally compatible graphical interface for this project (which is all but inconceivable given the variety of terminals in use on the existing network), it is interesting to note that the users generally prefer textual presentations as a more concise presentation of the practical knowledge they need. The original prototype embodied a picture of the devices under discussion, but this turned out to be much more pleasing to systems people than to the field staff; they preferred pictorial representations of the precise knowledge base instead, which would help them fix the problems more quickly. (The pictorial approach has been of value in other applications; for example, at some Ford plants portable PCs with standardized graphics screens are brought around to wherever assembly line robots are in need of attention. They offer displays such as "Put logic probes on *these* two pins; the meter should look like *this.*") As data speeds increase in the future, and terminals improve (hopefully converging to a standardized graphics interface), such pictorial components may well be added to the package, perhaps stored on CD-ROMs.

Future Prospects

A field technician training center in an adjacent province is already using this package during its in-house coursework; this exposure is arousing demand for field access there also. The same source code developed in this project is being used on a trial basis in similar fashion in four eastern provinces. In another province most field technicians already use portable terminals for other purposes. Soon they will all be using cellular phones, which could be plugged into the serial port of portable PCs, while portable printers will be able to reproduce hard copies of screen displays of system recommendations, so that they can be applied more conveniently to the problem at hand. Already, at least one repairman has phoned his manager from the top of a telephone pole, and asked him to log into the

system and talk him through the diagnostic procedure.

Although the literature of the expert system field strongly warns against trying to implement expert systems for new subject areas, the experience to date with development with the Level 5 shell encourages early applications to emerging technologies (such as fiber optics). In such domains, despite the absence of existing expertise, it appears feasible to sketch out a skeleton diagnostic system and flesh it out as experience accumulates.

CONCLUSION

This chapter has presented a snapshot of work-in-progress in establishing a valuable family of expert system modules at Bell Canada. One of the key features of this work has been its demand-driven nature. The relationship between the system builder and the user community has been flexible and cooperative, engendering rapid development of the systems, and fostering expansion into new areas. Two factors that seem to have influenced the successful insertion of expert systems technology into Bell Canada's ongoing operations are the modest, incremental goals of the project and the high productivity of the programmer's tools. The developer has to date deliberately chosen applications requiring relatively small knowledge bases but providing high payoffs. Expansion of the family of systems so far has been through the addition of further modules of similar scope and size, rather than the undertaking of larger, riskier applications. Such modules can be created rapidly using expert system shells, and therefore the domain experts, users, and managers can always see immediate benefits. By winning the trust of the corporate community in this way, the developer is now in a position to undertake larger and more diverse applications.

ACKNOWLEDGMENT

The author acknowledges the assistance of Alan N. Bloch and Alun D. Preece of the Center for Pattern Recognition and Machine Intelligence, Department of Computer Science, Concordia University, Montréal in the preparation of this paper.